

INTERACTIONS

 SUGGESTED TIME
30-45 MINUTES

OBJECTIVES

By completing this activity, students will:

- + explore different approaches to making projects interactive by solving a series of nine programming puzzles
- + gain more fluency in the concepts of conditionals, operators, and data, and the practice of testing and debugging

ACTIVITY DESCRIPTION

- ☐ On their own or in small groups of 2-3 people, challenge students to further explore Scratch by creating Scratch programs that solve each of the nine Interactions programming puzzles. These Interactions puzzles explore Sensing blocks, engaging some of the more advanced concepts in Scratch related to interactivity. Optionally, have the Interactions handout available to guide students during the activity.
- ☐ Each puzzle can have several possible solutions. Invite students or groups to share different solutions and strategies. We suggest the Pair-Share or Design Demo activity to allow students to share their work and describe their process. Optionally, have students add their projects to the Interactions studio or a class studio.
- ☐ Ask students to think back on the challenge by responding to the reflection prompts in their design journals or in a group discussion.

RESOURCES

- ☐ Interactions handout
- ☐ Interactions studio
<http://scratch.mit.edu/studios/487213>

REFLECTION PROMPTS

- + Which puzzles did you work on?
- + What was your strategy for solving the puzzles?
- + Which puzzles helped you think about your game project?

REVIEWING STUDENT WORK

- + Are the puzzles solved?
- + Did students explore other approaches for solving the puzzles?
- + Are there certain blocks or concepts students are still struggling with? How might you help?

NOTES

- + Choose particular challenges that highlight new blocks or concepts that you would like students to explore. Or let students invent their own interaction puzzle prompts.
- + Repurpose these puzzles as an unstructured activity for students who finish other activities early or as a warm-up challenge. Create a puzzle jar: print out, cut, fold, and place copies of each puzzle description in a jar. Then, let students pick puzzles from the jar to solve.

NOTES TO SELF

- ☐ _____
- ☐ _____
- ☐ _____
- ☐ _____

INTERACTIONS

WHAT DIFFERENTIATES A SCRATCH PROJECT FROM A STILL IMAGE OR A VIDEO?

Tackle these nine puzzles that engage some of the more advanced concepts in Scratch related to interactivity. Each of these challenges has several possible solutions.

START HERE

- ☐ Create a Scratch program for each of the nine interactivity puzzles.

FEELING
STUCK?

THAT'S OKAY! TRY THESE THINGS...

- ☐ Before getting started in Scratch, write down ideas in your design journal for possible ways of programming each of the interactivity puzzles.
- ☐ Work with a neighbor. Collaborating with a partner can be a great way to solve problems and gain new perspectives on ways of programming in Scratch!

☐ **PUZZLE 1:** Whenever you press the B key, the sprite gets a little bigger. Whenever you press the S key, the sprite gets a little smaller.

☐ **PUZZLE 2:** Whenever the sprite hears a loud sound, it changes color.

☐ **PUZZLE 3:** Whenever the sprite is in the top 25% of the screen, it says "I like it up here."

☐ **PUZZLE 4:** When the sprite touches something blue, it plays a high note. When the sprite touches something red, it plays a low note.

☐ **PUZZLE 5:** Whenever two sprites collide, one of them says: "Excuse me."

☐ **PUZZLE 6:** Whenever the cat sprite gets near the dog sprite, the dog turns and runs from the cat.

☐ **PUZZLE 7:** Whenever you click on the background, a flower appears at that spot.

☐ **PUZZLE 8:** Whenever you click on a sprite, all other sprites do a dance.

☐ **PUZZLE 9:** Whenever you move the mouse-pointer, the sprite follows but doesn't touch the mouse-pointer.

FINISHED?

- + Add each of the projects you create to the Interaction Studio: <http://scratch.mit.edu/studios/487213>
- + Help a neighbor!
- + Discuss your strategies for approaching each puzzle with a partner. Take notes about the similarities and differences in your methods.